# A METHOD OF PULSE-HEIGHT ANALYSIS SUITED TO SMALL COMPUTERS

Michael Hiles

A THESIS

in

The Department

of

Physics

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Physics
at Sir George Williams University
Montreal, Canada

# A METHOD OF PULSE-HEIGHT ANALYSIS SUITED TO SMALL COMPUTERS


## ABSTRACT

For purposes of analyzing gamma-ray spectral data in experiments at the Sir George Williams University Nuclear Physics Laboratory, a method using a transformation of the Gaussian function to a quadratic form for curve fitting procedures was applied to a PDP-8/L computer. Sample results as well as a discussion of techniques are included.

## ACKNOWLEDGEMENTS

The author would like to thank Mr. A. Khan and Mr. G. Balion for their assistance with the manipulation of the experimental equipment.

The author is very greatly indebted to Dr. A. Kipling for his help with the programming of the PDP-8/L computer. The ability to draw on his experience was invaluable in overcoming many difficult problems.

Most importantly, the author's sincere appreciation goes to Dr. N. Eddy for suggestion of this project, and more significantly, for his always positive attitude that was as solid as his scientific advice. Without his many good offices this project would have had doubtful success.

# TABLE OF CONTENTS

## 1. INTRODUCTION

Due to the random nature of the decay of the atomic nucleus combined with the nature of the design of the detection and counting systems, the data obtained in nuclear decay counting experiments are discrete in all the variables and take the form of statistical distributions. As emitted particles activate the detector, a pulse of a voltage corresponding in a linear fashion to the energy of the incoming particle is sent into the electronic counting system. An electronic memory location within the system is then incremented, the choice of location depending upon the pulse voltage, within limiting values. Thus if particles of only one particular energy are encountered then one particular memory location will be incremented once for each particle encountered. The compliment of memory locations is arranged in such a way that low energy particles cause memory locations that are assigned low identification numbers to be incremented, and vice versa.

Incident photons would cause an increment in a memory location, hereafter called a channel, for each particle predicted from theoretical considerations. However due to the nature of the electronic counting systems the contents of the channels about the expected channel also grow and what results is a normal distribution about some mean.

It is the present purpose of this work to demonstrate a method of determining the important parameters of the data in a fashion well suited to the Nuclear Physics Laboratory at Sir George Williams University.

The parameters required to extract information from the data of the experiments designed to determine energy and activity of an emmitter of Gamma radiation appear as constants in a Gaussian distribution

In the present method, which is a curve fitting technique, a transformation is applied to the Gaussian so as to avoid the necessity of using non-linear curve fitting procedures, thus making application more practicable to small computers.

A program was written utilizing the Gauss Method of Least Squares for the case of a second degree polynomial to fit data supplied directly from the computer core memory to a polynomial of second order. The background radiation and detector noise were approximately determined by using an exponential function in order to correct the raw data. The resulting corrected data were put in the form of the elements of a square matrix, those elements being the coefficients of three simultaneous linear equations. A column matrix was also constructed whose elements were the the constants of the three equations. From the solutions the constants of the Gaussian function were calculated. The computer output was contingent upon the closeness of the width of the function, given by the quantity σ, to an estimate provided by the user. The program was designed to allow the user to choose the various sections of the data to be analyzed.

## Previous Techniques

### 2.1. Non-Linear Method

The method used at Sir George Williams University for fitting data to a Gaussian function,

$$Y = C \exp(-(x-\mu)^2 / 2\sigma^2) \qquad (2.1)$$

is the general method for fitting data to a non-linear function. A function $f(x)$ is viewed as having its parameters acting as independant variables. With this as a basis the function is expanded in a generalized Taylor's series about the parameters $a_1, a_2, \ldots, a_n$ of $f(x)$. Only the terms in the first order of smallness are used in this method. This result is linear and may be fitted to data using the Gauss Method of Least Squares.

Consider

$$Y = f(a_1, a_2, \ldots, a_n, x) \qquad (2.2)$$

where the $a_i$'s enter in a non-linear fashion. The true residuals for each $y_i$ observed for a known $a_i$ are

$$r_i = f(a_1, a_2, \ldots, a_n, x_i) - y_i \qquad (2.3)$$

where $f$ is the function of 'best fit' through the data.

If we can obtain an approximation for the $a_i$'s then we may compute an approximate residual:

$$R_i = f(a_1^0, a_2^0,,,, a_n^0, x_i) - y_i \qquad (2.4)$$

for n data pairs. The problem is to obtain an improved approximation for the $a_i$'s.

Let us expand the function $f(a_1,,,x)$ about the $a_i^0$'s using a Taylor's series:

$$f(a_1,,,x) = f(a_1^0,,,x) + \frac{\partial f}{\partial a_1}(a_1 - a_1^0) + \ldots$$

$$+ \frac{\partial f}{\partial a_n}(a_n - a_n^0) \qquad (2.5)$$

Now let

$$\delta a_i \equiv a_i - a_i^0 \quad ; \quad \frac{\partial f_i}{\partial a_i} \equiv \left.\frac{\partial f}{\partial a_i}\right|_{x = x_i,\ a_i = a_i^0} \qquad (2.6)$$

If we now subtract $y_i$ from both sides of (2.5), after some rearrangement we obtain

$$r_i = R_i + \frac{\partial f_i}{\partial a_i}\delta a_1 + ,,, \frac{\partial f_i}{\partial a_n}\delta a_n \qquad (2.7)$$

We now use the well known results of the Method of Least Squares[1] and obtain a system of n equations for n parameters. In matrix form this is

$$
\begin{bmatrix}
\sum_{1}^{n}(\partial f_i/\partial a_1)^2 & \sum\frac{\partial f_i}{\partial a_2}\frac{\partial f_i}{\partial a_1} & \cdots \\
\vdots & & \\
\cdots & \cdots & \sum(\partial f_i/\partial a_m)^2
\end{bmatrix}
\begin{bmatrix}
\delta a_1 \\ \delta a_2 \\ \vdots \\ \delta a_m
\end{bmatrix}
=
\begin{bmatrix}
\sum R_i\frac{\partial f_i}{\partial a_1} \\ \vdots \\ \sum R_i\frac{\partial f_i}{\partial a_m}
\end{bmatrix}
\qquad (2.8)
$$

If any $\delta a_i$ is larger than desired the process can be repeated as an iteration.

The theoretical criticism of this method is that it provides an approximation to what is in the first place an estimation. It would be of much more use to have a method that does not require iterations. This is in general not possible but can, in some cases, be realized by choice of a Frobenius series which does not require the use of the Taylor's series expansion in the previous discussions. Thus it should be tried to linearize the function $f(a_1,\ldots,a_n,x)$.

A rather more computational disadvantage is the complexity of the numerical calculations required in the non-linear method. This method as applied to the Gaussian function is employed at the Nuclear Physics Laboratory of Sir George Williams University and uses a large percentage of the available core and up to ninety seconds of central processor time on a CDC 6400 computer, a remarkably large usage particularly for a scientifically oriented machine. Alsq, at Sir George Williams University the CDC 6400 is physically remote from the experimental equipment necessitating the preparation of data on magnetic tape, which then has to be manually carried to a seperate site to be mounted on a tape drive at the computer center, which are in short supply. Consequently better methods were looked for and then implemented.

## 3. MATHEMATICAL TECHNIQUE

### 3.1. Linearization Method

As was previously pointed out the Gaussian function,

$$Y = C \exp(-(x-\mu)^2/2\sigma^2) \quad , \qquad (3.1)$$

being non-linear, does not lend itself to simple techniques for curve fitting, using the criterion of Least Squares. Consequently some technique of simplifying the mathematics was sought.

It was decided to take the criterion of Least Squares as fundamental; by the measure of the $\chi^2$ test it can be shown that of the various other methods of curve fitting the Method of Least Squares provides the "best fit",[2] and thus the most reliable values for related energy and intensity data.

In the light of the great simplification that results from choosing a linear or power series function to be fitted to the test data, combined with the special requirement of relatively straightforward application to the PDP-8/L computer that is hardwired into the Sir George Nuclear Physics Laboratory, it was felt that some method of expressing the Gaussian function in the above-mentioned form should be tried.

The resulting mathematical arguments took the following form:

$$Y = C \exp(-(x-\mu)^2/2\sigma^2) \qquad (3.1a)$$

where $Y = y - \beta$ ; $\beta$ = background correction factor.
The y is the original data. Thus we have

$$y - \beta = C \exp(-(x-\mu)^2/2\sigma^2) \tag{3.2}$$

Squaring both sides gives

$$(y - \beta)^2 = C^2 \exp(-(x-\mu)^2/2\sigma^2)^2 \tag{3.3}$$

Now making a logarithmic transformation,

$$\ln\{(y - \beta)^2\} = \ln\{C \exp(-(x-\mu)^2/2\sigma^2)\}^2 \tag{3.4}$$

Now let

$$\ln\{(y - \beta)^2\} = Z, \tag{3.5}$$

and applying the usual rules of logarithims
we have,

$$Z = 2\ln\{C \exp(-(x-\mu)^2/2\sigma^2)\} \tag{3.6a}$$

$$Z = 2 \{\ln C + (-(x-\mu)^2/2\sigma^2)\} \tag{3.6b}$$

or

$$Z = -(x-\mu)^2/\sigma^2 + (2 \ln C) \tag{3.6c}$$

Expanding the first term on the right gives:

$$Z = (-1/\sigma^2)(x^2) + (2\mu/\sigma^2)(x) + \{2 \ln C - (\mu^2/\sigma^2)\} \tag{3.7}$$

which is of the form

$$s = a_2 t^2 + a_1 t + a_0$$

and applies directly to the straightforward technique of the Method of Least Squares, thus avoiding all the above-mentioned drawbacks of the technique for handling the general non-linear case. Thus we have,

$$\begin{aligned} a_2 &= (-1/\sigma^2) \\ a_1 &= (2\mu/\sigma^2) \\ a_0 &= (2 \ln C + (\mu^2/\sigma^2)) \end{aligned} \tag{3.8}$$

from whence the parameters of the Gaussian are uniquely determined, and thus after comparison to a known reference, the energy and intensity of the beam of Gamma photons is available.

The advantages of this technique are that the best estimate of the function to be fitted is given exactly with no need for iterations thus reducing computer time. Calculations are also made simpler so as to avoid software generated algorithmic failures. Moreover stability questions due to iteration methods are no longer required.

The origens of the quantity $\beta$, the background correction factor, are as follows. The effects of background and other unknown effects around the experiment, detector efficiency and Johnson's noise in the detector is approximately a decaying exponential in form. That is, $\beta$ is represented by

$$\beta = C_0 \exp(kx) \tag{3.9}$$

where $C_0$ is the amplitude of the background effects at channel zero and k is the decay factor. By the same arguments as in 3.2 we have

$$\ln\beta = kx + \ln C_0 \tag{3.10}$$

Letting

$$\gamma = \ln\beta \; ; \quad \phi = \ln C_0 \tag{3.11}$$

produces

$$\gamma = kx + \phi \tag{3.12}$$

which again may be fitted from data in the same simple linear fashion by the Method of Least Squares.

Thus the final working equation to be fitted is written as:

$$\ln\{(y - (C_0 \exp(kx))\} = (-1/\sigma^2)(x^2) + (2\mu/\sigma^2)(x) + \{2 \ln C - (\mu^2/\sigma^2)\} \quad (3.13)$$

The formalism for applying the Method of Least Squares is available in any book on numerical analysis.[3] The calculations result in, for our case, three equations for the three unknowns $a_2$, $a_1$, and $a_0$. From these values the quantities C, β, and σ follow immediately. The equations for use in the numerical calculations are, in matrix form,

$$\begin{bmatrix} n & \Sigma x_i & \Sigma x_i^2 \\ \Sigma x_i & \Sigma x_i^2 & \Sigma x_i^3 \\ \Sigma x_i^2 & \Sigma x_i^3 & \Sigma x_i^4 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \Sigma y_i \\ \Sigma x_i y_i \\ \Sigma x_i^2 y_i \end{bmatrix} \quad (3.14)$$

where the summations are over the n data pairs $(x_i, y_i)$.

Using the solution of this matrix equation and equation 3.8 we obtain the required information.

# PROGRAMMING TECHNIQUES

## 4.1. General Considerations

To do the calculations associated with the technique previously described a program was written to run on a PDP-8/L computer that is owned by the Physics Department of Sir George Williams University. The reasons for the choice of this computer were two-fold; firstly this unit is hardwired into the multichannel analyzer in the Nuclear Physics Laboratoty, and secondly, the machine is "dedicated" and consequently is available at any time and at no cost. The first of these reasons is the most important. By having the computer connected directly to the experiment the data, collected in the memory of the multichannel analyzer, is transfered to the core of the computer by a program written specifically for the purpose[4]. This program, which takes less than one second to run, transfers two thousand and forty eight discrete pieces of information automatically. Thus it follows that this machine should be pressed into use to carry out the Least Squares calculation.

The program was designed to solve the three simultaneous linear equations generated by the Method of Least Squares and provide the values of C, $\sigma$, and $\mu$. To do this it was divided into a sequence of logical steps. The program listing given in the appendices should be consulted.

### 4.2. Data Conversion

The data is transferred from the 24-bit Multichannel Analyzer (MCA) core memory in binary-coded-decimal format. The contents of each channel of the MCA is represented by a 24-bit BCD number, the largest possible number being 999999, in decimal format, so the data from each channel occupies two 12-bit computer core memory locations. The 24-bit BCD number is converted first to a double precision binary number occupying two 12-bit memory locations, and then to a floating point number occupying three 12-bit memory locations. Information about the floating point number format is given in reference 6.

The conversion from BCD to binary to floating point format is done in locations $0304_8$* to $0324_8$. The most significant portion of the BCD number $y_i$ in location (I)A in the second of the PDP-8/L's two 4096-word memory fields is converted to a binary (BIN) number using the FCON pseudo-instruction created for this purpose and then to a floating point number using the FLOT instruction from the Floating Point Package. It is then multiplied by one thousand and is stored in the buffer register. The (I) indicates indirect addressing (see reference 6).

*$nnnn_8$ are octal numbers, a set of integers generated according to a modulo (7) rule.

The least significant portion of $y_i$ is the contents from memory location (I) A in memory field one, A being incremented by one beforehand. This number is converted to BIN form as before, then changed to floating point form, the most significant portion of $y_i$ is then added to it from the buffer and the result is then placed in the three word floating point memory location designated (I) STORE. The contents of (I) STORE contain the floating point binary number $y_i$ equivalent to the BCD number generated in the twenty-four bit memory location of the multichannel analyzer.

This sequence is repeated, with appropriate increment of storage pointers and counting locations until the n data points $y_i$ have been converted and stored for use in the computations. The program then proceeds to calculate the background effects for the quantity β.

4.3. <u>Calculation of β</u>

The background β is calculated by fitting an exponential function to the data pairs $(x_i, y_i)$, $(x_{i+1}, y_{i+1})$, $(x_{i+2}, y_{i+2})$, $(x_{i+n-2}, y_{i+n-2})$, $(x_{i+n-1}, y_{i+n-1})$, $(x_{i+n}, y_{i+n})$ using the technique described in the section on Mathematical Techniques. The program parts from location $0337_8$ to $0503_8$ perform this function. The quantities $\sum_1^n x_i$, $\sum_1^n x_i^2$, $\sum_1^n y_i$, $\sum_1^n x_i y_i$, and n are calculated from the data stored in the locations labeled (I) STORE and accumulated in the locations SUMX, SUMX2, SUMY, SUMXY and N. These values are used as the elements of the determinants of the matrix equations for Cramer's Rule.

The results of these calculations are the coefficients $a_0$ and $a_1$ of the general equation

$$Y = a_1 x + a_0 \tag{4.1}$$

where

$$a_1 = k \quad ; \quad a_0 = \phi \tag{4.2}$$

from equation (3a).

The exponential of $a_0$ is then stored in the memory location labeled C and $a_1$ is stored in location K. These values, then, are to determine the characteristics of the background for the purposes of applying corrections to the data obtained in the Conversion section of the program.

4.4. Main Coefficients

The program sections from $0526_8$ to $0623_8$ calculate the elements of the 3X3 determinants generated by the mathematical method. The values of $y_i$ used are the corrected values $Z_i$ from section 3.1. These are produced by taking

$$B = C_e \exp(kx) \tag{4.3}$$

using the values stored in locations C and K, negating this result, and adding to it the converted $y_i$

from the (I) STORE location. This is put into BUF2 and the floating point $x_i$ is put into BUF1. Using the values of BUF2 and BUF1 the quantities $\Sigma x_i$ (SUMX), $\Sigma x_i^2$ (SUMX2), $\Sigma x_i^3$ (SUMX3), $\Sigma x_i^4$ (SUMX4), $\Sigma Z_i$ (SUMY), $\Sigma Z_i x_i$ (SUMXY), and $\Sigma Z_i x_i^2$ (SUMX2Y) and accumulated in the locations shown in brackets by incrementing the values of X and (I)STORE n times. After n passes the program jumps out into the routine that calculates the determinants for $a_2$, $a_1$, and $a_0$.

4.5. <u>Determinants of Coefficients</u>

From the elements created in the last section values of the four determinants for the application of Cramer's Rule are calculated. These functions are performed from locations $0624_8$ to $0672_8$ and from $2000_8$ to $2140_8$, the gap being due to the need to accomodate the ODT-8 debugging software which is used for editing purposes as well as a running monitor. The determinants are calculated in the usual way by expanding the cofactors associated with the first row of elements. The value of the determinants is then stored in the memory locations labeled DETM, DETA1, DETA2, and DETA3, which are the values of the principle determinant, and those associated with $a_0$, $a_1$, and $a_2$.

4.6. <u>Calculation of Gaussian Parameters</u>

Memory locations $2141_8$ to $2220_8$ calculate the parameters C, $\mu$, and $\sigma$ from the values stored in DETM, DETA1, DETA2, and DETA3 using the Cramer relations

$$
\begin{aligned}
a_2 &= \mathrm{DETA3/DETM} \\
a_1 &= \mathrm{DETA2/DETM} \qquad (4.4) \\
a_0 &= \mathrm{DETA1/DETM}
\end{aligned}
$$

and using the relations developed in the section on Mathematical Techniques the parameters become,

$$
\begin{aligned}
\sigma &= \{-1/(\mathrm{DETA3/DETM})\} \\
\mu &= \{(\mathrm{DETA2/DETM})\sigma^2\}/2 \qquad (4.5) \\
C &= \exp\{((\mathrm{DETA1/DETM}) - (\mu^2/\sigma^2))/2\}
\end{aligned}
$$

where $\sigma$ is stored in DETA3, $\mu$ is stored in DETA2 and C is stored in DETA1.

## 4.7. Supplementary Software

In memory locations $4200_8$ to $4234_8$ the software associated with the pseudo-instruction FCON operates. This software is designed to convert a twelve-bit BCD word to a twelve-bit binary word and is a standard piece available from the Digital Equipment Corporation mentioned previously. Each time the computer encounters the FCON pseudo-instruction it jumps into this routine, performs the conversion function and returns to the main program.

The locations $4235_8$ to $4253_8$ are a simple routine to perform two line feeds on the teletype so as to separate the output peak by peak.

### 4.8. Output

Memory locations $2221_8$ to $2251_8$ test the value of $\sigma$ against a set test value labeled TEST and if the $\sigma$ is larger than TEST no output is performed. This corresponds to assuming a peak has been fitted correctly if its width is less than some estimate provided earlier in the program.

### 4.9. Initializations

The remaining sections of the program perform initializations of counters, intermediate counters and accumulators are set to zero and incremental operations are done. In a normal run the switch register is set to $0200_8$ and the computer is started. The program then sets its own values in the main initialization. However if the switch register is set to $0000_8$, the program halts at appropriate spots to allow the user to set values for 1) the number n of channels to be used for each fit, 2) the value of TEST as discussed previously, 3) the first channel to be used in the fitting procedures and 4) the number of tries at making a good fit by incrementing X and its related quantities. After these values have been entered from the teletype the program continues with these new values of initialization.

4.10. Program-Related Calculation Techniques

The method used to find solutions to the simultaneous equations that result from the Method of Least Squares was the method of determinants known as Cramer's Rule. There may be valid objections raised as to the use of this technique since in the event that the principle determinant approaches or becomes zero, the solutions become unreliable due to computational inaccuracies or undefined, as the case may be.

This would suggest the use of such procedures as Gauss-Jordan elimination, or matrix inversion. It has been found from practical experience that the principle determinant does not approach zero; on the contrary it sometimes reaches the order of $10^{26}$ and higher, thus the objection to the use of Cramer's method does not apply in this case. The advantages of Cramer's method on the other hand, are much greater speed and better use of computer core memory.

Possibilities of resorting to the Method of Orthogonal Polynomials was considered but discarded as being too cumbersome to be in the spirit of the stated purposes. Techniques, such as Forsythe's Method[5], while being extremely accurate and stable, are designed to reduce hazards that are not encountered in the present situation. Consequently Cramer's Method was adhered to.

## 4.11. Exterior Software

Two pieces of software supplied as a part of the PDP-8/L computer package are used externally to the program. These are the Floating Point Package (FPP) and the Octal Debugging Technique (ODT).

The Floating Point Package is a program consisting of a series of subroutines that perform arithmetic operations, certain unary operations such as exponential and trigonometric functions, and utility operations such as conversion from floating point format to integer format and vice versa. There are also input and output routines using the teletype in either fixed decimal or power-of-ten notation (fortran F or E notation). This software has its own set of pseudo-instructions that is listed at the beginning of the programs in the appendices I, II. The need of the FPP can be appreciated when one considers that the basic instruction set that operates the hardware of the PDP-8 series computers allows binary arithmetic as its only operation. Furthermore the twelve digit binary number that is the machine's basic quantity may have at first sight a maximum value of 4096. However the machine recognizes the numbers 2049 to 4096 as negative numbers going from -2047 to -1, consequently the maximum number the computer will accept for strictly hardware arithmetic is ±2048, with only integer values between ±2048. The FPP, however, allows arithmetic on real numbers from 0 to approximately

$\pm 10^{610}$ for one version supplying six significant figures of accuracy, and 0 to $\pm 10^{31}$ with seven figures of accuracy using a second version of the FPP. Calculating the two portions seperately, complex numbers may also be accommodated. The Floating Point Package is for all intents and purposes indispensable.

The second piece of software used is the Octal Debugging Technique (ODT), so called because its primary use is to debug programs that are in the core of the computer. Its value in the present situation is centered around its function in placing a breakpoint in the program. The breakpoint when placed in the program stops the program at that point and returns control to the ODT and the ODT awaits further instructions from the user. Consequently for present purposes the breakpoint is set at the end of the program. The instruction nnnnG is then given from the keyboard, where nnnn is the starting address of the program, and control is given to the program until the breakpoint is encountered (at the end of the program) and then the ODT takes over once again. The process may be repeated at will and thus the ODT software in this application is an automatic operating system.

# 5. ELECTRONICALLY GENERATED SPECTRA

## 5.1. Pulser Peak Program

There are occasions when data is to be created within the experimental apparatus to be used for purposes such as calibration and checking the system. It was consequently required to develop a method of dealing with this data to obtain the peak centroids. Using the data points $(x_i, y_i)$ the centroid, W, is given by,

$$W = \Sigma x_i y_i / \Sigma y_i$$

The most obvious property is the zero background. Thus it is not necessary to use a technique to subtract a non-existant background. The independance of the peaks, that is, the zeros between the peak data is used to signify the end of the individual peak and instigate output. On scanning through the field of data, when a non-zero value of $y_i$ is encountered the program begins to accumulate the values $\Sigma x_i y_i$ and $\Sigma y_i$. When another zero is encountered from the $y_i$'s the program calculates the value of the centroid W and prints the result. After the program has sampled 1024 channels of data the program halts the computer. The method is reliable, the values produced are trustworthy for the present purposes, and the program is fast, taking about six seconds of central processor time to handle 1024 channels of data. A program listing is in the appendix II.

# 6. RESULTS AND DISCUSSION

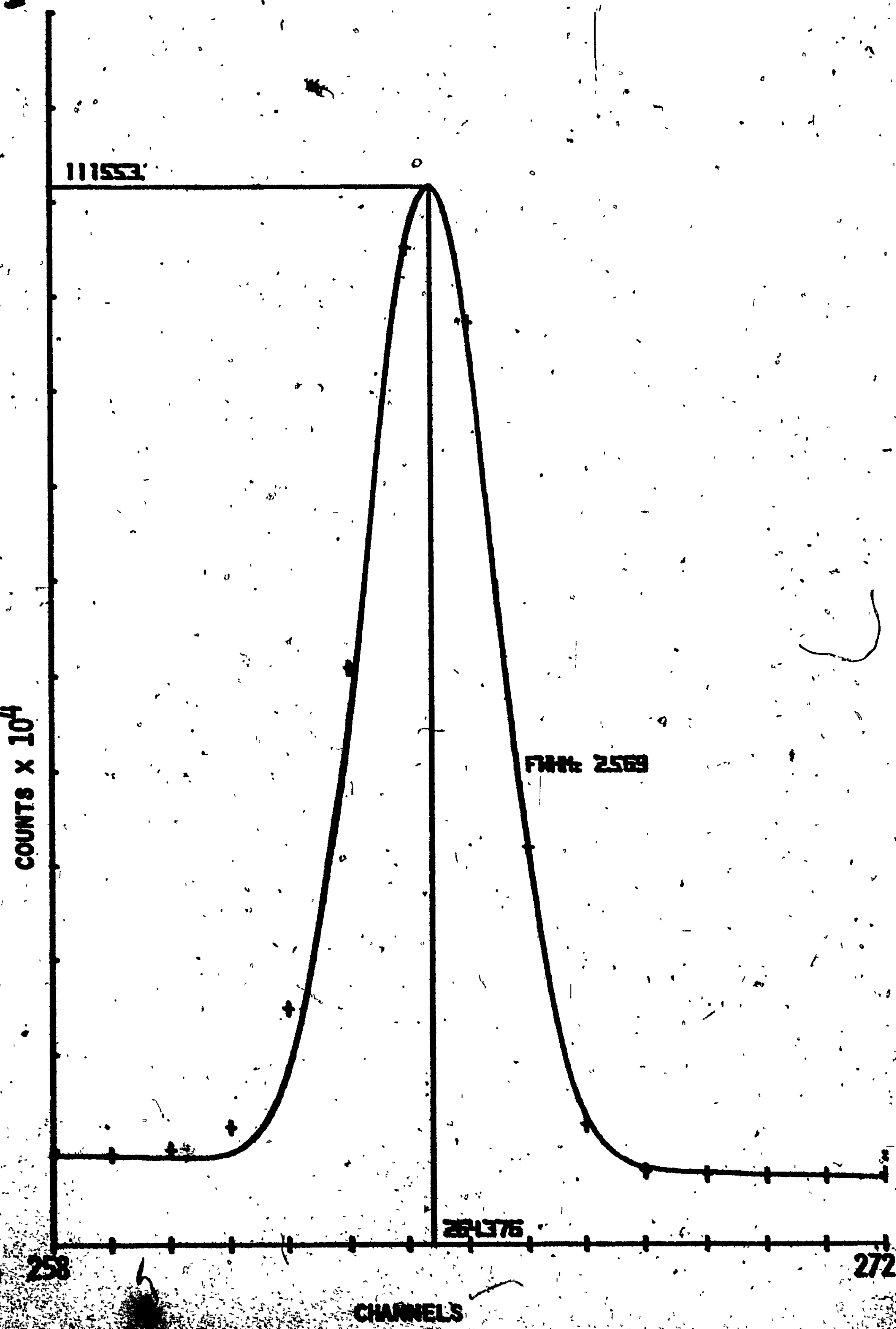
## 6.1. Method of Linearization

As a practical test of the Quadratic method a γ - ray spectrum of radium was investigated with a view to making some comparisons with the results produced by the non-linear technique previously discussed. As pointed out, this method is employed on a CDC 6400 computer and has been shown to provide very reliable results. Consequently results of the present method were compared as far as possible. Certain differences between the results given are to be expected since the programs using the methods in question are not designed to do exactly the same thing. Considering the intended use of the order 2 method applied to the PDP-8/L computer is to find the peak centriod, the most important quantity is the parameter μ, which is related to particle energy. The following Table I is the results of the two methods applied to peaks of a radium spectrum. Peaks 1 and 2 are X - ray peaks and 8 is a weak γ peak that is very often statistically poor as compared with the other five γ peaks, which are the principle peaks found in radium. This weak peak is observable due to relatively long counting times used to produce good statistics and signal-to-noise ratios.

FIGURE I

**Least Squares Fit Using**

**Linearized Gaussian Function.**



111553.
FWHM: 2.569
264.376
258
272
COUNTS X $10^4$
CHANNELS

| Peak Number | μ(Channel) | |
|---|---|---|
| | Quadratic | Non-Linear |
| 1 | 38.249 | 38.347 |
| 2 | 47.511 | 47.510 |
| 3 | 128.284 | 128.357 |
| 4 | 174.005 | 174.088 |
| 5 | 217.772 | 217.780 |
| 6 | 264.376 | 264.327 |
| 7 | 475.870 | 475.919 |
| 8 | 896.544 | 896.632 |

**TABLE I**

Values of μ for Peaks of a γ Spectrum

The average difference between the peak values μ given by the two methods is .055 channels and the associated standard deviation is .035 channels.

In the next table the values of σ that were obtained have been converted to the quantity "Full Width at Half Maximum" (FWHM) by the formula

$$\text{FWHM} = 2\sqrt{2\ln 2}\,\sigma$$

for comparison purposes.

| Peak Number | FWHM(Channel) Quadratic | Non-Linear |
|---|---|---|
| 1 | 3.80 | 3.87 |
| 2 | 3.17 | 3.62 |
| 3 | 2.67 | 2.70 |
| 4 | 2.71 | 2.73 |
| 5 | 2.75 | 2.71 |
| 6 | 2.57 | 2.71 |
| 7 | 2.81 | 2.87 |
| 8 | 3.36 | 3.29 |

TABLE II

Values of FWHM for Peaks of a

γ Spectrum

In the next table the values for C are listed. They are included for completeness but are not used for comparison purposes since these values are for peaks without the background, which is what is given by the program output, and does not correspond to any value listed by the non-linear program.

It should be noted that these values are particularly vulnerable to computer generated errors in calculating and consequently the computing system should be capable of maintaining as much accuracy as possible.

| Peak Number | C (Counts) |
|---|---|
| 1 | 84451 |
| 2 | 24708 |
| 3 | 25823 |
| 4 | 35505 |
| 5 | 67925 |
| 6 | 103296 |
| 7 | 59495 |
| 8 | 7723 |

**TABLE III**

**Amplitudes for Peaks of a $\gamma$ Spectrum**

The differences in the values listed may be due to a number of factors. The non-linear program called SPED uses a different technique for subtracting background; it considers the background a straight line and subtracts accordingly while in the order 2 program written for the PDP-8/L the background is considered a decaying exponential, which seems to be more consistent with the facts.

It should also be noted that the differences may be attributed to the fact, that the non-linear method uses a Taylor's series taken only to first order of smallness and consequently the functions are in fact not the same. When one keeps in mind that the experimental data may also vary from a true normal distribution, certain discrepancies may be understood.

### 6.2. Pulser Peak Program

The uses of the pulser peak program will be mainly for testing the experimental equipment with respect to a known source. That source is, in this case, an ORTEC RESEARCH PULSER designed for the purpose.

As a trial for the program a spectrum of generated pulser peaks was generated by incrementing the pulse height of the pulser output in exactly equal amounts. Giving the pulser output arbitrary values of 1, 2, 3, etcetera, the table of results is as follows:

| Pulser Output (In Arbitrary Units) | Peak Centroid (Channel) |
|---|---|
| 1 | 4.730 |
| 2 | 172.000 |
| 3 | 340.000 |
| 4 | 508.000 |
| 5 | 676.000 |
| 6 | 844.986 |

**TABLE IV**

**Centroids of Equally Spaced Pulser Peaks**

The average difference between the peaks is 168.051 channels with an associated standard deviation of 0.611 channels. The data is based on frequencies at the peak of around $10^4$ and it is consequently felt that the peak values are reliable. The program's calculation routines were checked against a Hewlett-Packard HP-45 calculator and were found to be reliable. The tabulated results would suggest that reasons for the non-linearity at low energies (channels) and high energies should be investigated.

## OPERATING INSTRUCTIONS

As with any machine calculating procedure there is a set order of operation to make this analysis program work. All instructions refer to devices and controls readily visible on the experimental equipment or on the PDP-8/L computer.

### 7.1. Data Transfer From The Multichannel Analyzer To The PDP-8/L Memory

There is a piece of software in existance that runs on the PDP-8/L computer and is designed to execute data transfer from the memory of the multichannel analyzer (MCA) to the computer core. It is labeled MCA SERVICE PROGRAM and is available in the Nuclear Physics Laboratory at Sir George Williams. This program must be placed in the computer with the help of the binary loader software, which can be assumed to be always in the memory. If the loading procedure does not operate reference 6 should be referred to. The transfer program is loaded into memory from a paper tape reader on the teletype of the computer. This program is placed in the second of the computer's two 4K memory fields, and the loading sequence is as follows:

1) Turn on computer power switch and teletypr.
2) Set Switch Register (SR) to $7777_8$
   ($111\ 111\ 111\ 111_2$)
3) Place MCA SERVICE PROGRAM paper tape in

paper tape reader and set to START.

4) On console set INST FIELD to 0 and DATA FIELD to 1.

5) On console press LD ADDR (Load Address).

This sets the starting address of the MCA SERVICE PROGRAM.

6) On console press START.

The paper tape will begin reading into the memory. When the whole program is loaded the computer will halt.

The MCA SERVICE PROGRAM may now be run in conjunction with the MCA. The sequence is as follows:

1) On MCA set MODE switch to DISPLAY.
2) On MCA set MEMORY switch to appropriate setting.
3) On MCA set READ IN/OUT switch to PRINTER.
4) On MCA set MODE switch to STOP.
5) On MCA plug transfer cable into back of unit.
6) On MCA set MODE switch to READ IN/READ OUT.
7) On PDP-8/L set INST FIELD and DATA FIELD both to 1.
8) On PDP-8/L set SR to $0310_8$ ($000\ 011\ 001\ 000_2$)
9) On PDP-8/L press LD ADDR, then START.

The teletype will activate momentarily and after about one second,

10) On console press STOP.

The data is now stored in the computer's memory.

11) On MCA set MODE switch to STOP.

12) On MCA unplug cable at rear of unit.

13) On MCA set MODE switch to DISPLAY.

This finishes the transfer process and places 1024 channels of data into the appropriate section of memory for both the present programs.

7.2. Loading The Analysis Software

The loading of the analysis program and attendant software is somewhat different from the previous loading sequence. These programs, due to thier length, are stored on magnetic tape (magtape). The paper tape program that goes with them is a loading program called READER which must be loaded into memory as before.

1) On console set and leave DATA FIELD and INST FIELD switches at 0.
2) Proceed as for MCA SERVICE PROGRAM.

Using the READER program the analysis program and software may be loaded from magtape as follows:

1) Mount software magtape on tape drive and press LOAD FORWARD twice. Then press ON LINE.
2) On console set SR to $4000_8$ ($100\ 000\ 000\ 000_2$).
3) Press LD ADDR, then START.
4) Set SR to $2000_8$ ($010\ 000\ 000\ 000_2$) and press CONTINUE.

5) Set SR to $0000_8$ ($000\ 000\ 000\ 000_2$) and press CONTINUE.

Tape will move.

6) Press STOP.

7) Set SR to $4000_8$, press LD ADDR, then START.

8) Set SR to $3400_8$ ($011\ 100\ 000\ 000_2$) and press CONTINUE.

9) Set SR to $4400_8$ ($100\ 100\ 000\ 000_2$) and press CONTINUE.

Tape will move.

10) Press STOP.

The analysis program and software will now be in the core ready for use.

## 7.3. Operating The Analysis Program

The analysis program uses the ODT as an operating device for interaction use with the teletype, removing the need for unnecessary use of the computer console. Use requires the following procedures:

I) On the console set SR to $1000_8$ ($001\ 000\ 000\ 000_2$).

II) Press LD ADDR, then START.

III) Type 2271B to set breakpoint.

The system is now ready for active use. To run the analysis, the user must decide whether to choose his parameters or use those provided by the program. The result of the decision requires the following:

a) If the user wishes to control the analysis parameters the SR should be set to $0000_8$.

b) If the user wishes the program to run independantly the SR should be set to anything but $0000_8$.

On the teletype the user then types 200G. Depending on the previous decision the program will a) stop or b) analyze the whole spectrum and output data.

If a) has been chosen, the program will stop and wait for input, in the following sequence:

1) Type 200G

The computer stops.

2) Type n for the number of channels to be analyzed.

The computer stops.

3) Type value of TEST, the maximum value of $\sigma$ allowed.

The computer stops.

4) Type the value of X, the starting channel number.

The computer stops.

5) Type the value of TALLY, the number of increments to be executed with a fit for each increment.

The computer then proceeds to calculate on the basis of the parameters set. In the case of either a) or b), when the program is finished and results are printed, control reverts to the user through the functions and commands of the ODT, and the user may run an analysis again by typing 200G.

7.4. Use Of Pulser Program

The use of the pulser program is slightly different from the analysis program. The program and Floating Point Package are loaded from paper tape in the same manner as the program READER, the method being well documented in reference 4, which is a computer user's manual.

To run the pulser program the user sets the SR to $0200_8$ ($000\ 010\ 000\ 000_2$) and presses LD ADDR, and then START. When the output is finished and the program has stopped the program may be run again by repeating the sequence.

## ANALYSIS TECHNIQUES

### 8.1. Use Of The Main Program

The strength of the main program is to make preliminary analyses of individual peaks in a spectrum. The program called SPED using the non-linear techniques previously mentioned and running on the CDC 6400 computer has been shown to provide excellent results and is useful for analyzing complete spectra. If, however, data is wanted immediately on certain peaks in a freshly generated spectrum, the present program is ideal. The actual use of the program requires some practice to be used most efficiently.

The four parameters of analysis are of great importance in determining the quality of the data generated. The parameters are:

1) n, the number of channels to be analyzed.
2) TEST, the maximum value of $\sigma$ allowed.
3) X, the starting channel of the analysis.
4) TALLY, the number of times the basic X shall be incremented for a new fit.

Of these parameters the first is the most important. This parameter decides how much data is used in the calculations, that is whether the flat portions of the peak shall be included or not. Due to the mathematics this is of importance; the amount and symmetry of the data effect the relative weightings of the data and consequently effect the fit produced. Thus a number of tries, varying parameters, are sometimes in order.

The pulser peak program has no such parameters to vary. Its simple calculating technique does not require any choice on the part of the user. For use, it is sufficient to start the computer and wait for output, which will occur in less than ten seconds, if there is any data to form a pulser peak.

## CONCLUSION

Summing up, the tests performed using the program utilizing the present technique produced results in accord with the design objectives. Since the main program was based on a mathematical technique introduced for simplification, the results demonstrate the validity of the present method which uses the Gaussian function. Moreover it clearly shows its practicability for the situation at the Nuclear Physics Laboratory at Sir George Williams University.

The present program, while successful, should not be considered final. Within the limits of the PDP-8/I computer, routines could be set up to test the quality of fit and make decisions to change analysis parameters so as to optimize results.

Apart from its use in the present situation the mathematical method is of use for any set of data having a normal distribution and may prove useful in other applications.

# APPENDIX I

## Linearized Gaussian Fit Program

```
                FIXMRI   FJMP=0000
                FIXMRI   FJMS=7000
                FISZ=0000
                FEXT=0000
                FSQU=0001
                FSQR=0002
                FSIN=0003
                FCOS=0004
                FATN=0005
                FEXP=0006
                FLOG=0007
                FNEG=0010
                FIN=0011
                FOUT=0012
                FFIX=0013
                FLOT=0014
                FNOR=7000
                FCDF=7001
                FSW0=7002
                FSW1=7003
                FHLT=7004
                FSMA=7110
                FSZA=7050
                FSPA=7100
                FSNA=7040
                FNOP=7010
                FSKP=7020
                FCON=0015
                PAUSE

                *1
0001    2000    REPP,REP
0002    4235    CRLFP,CRLF
0003    0000    TEST,0
                *5
0005    0004    TESTP,4
0006    2000    APP,2000
                *10
0010    0000    STORE,0
                *20
0020    0000    BUF1,0;
0021    0000    0;
0022    0000    0
0023    0000    A,0
0024    0000    CTR,0
0025    0000    CTR2,0
0026    0001    N,1
0027    0000    X,0
```

```
0030  0000  ZERO,0;
0031  0000  SUMXY,0;
0032  0000  0;
0033  0000  0;
0034  0000  SUMX2,0;
0035  0000  0;
0036  0000  0
            *63
0063  0000  SUMX,0;
0064  0000  0;
0065  0000  0
0066  7775  NEG3,-3
0067  7772  NEG6,-6
0070  0036  NP,36
0071  0000  N,0
0072  2002  AP,2002
0073  2275  STOREP,BEGINP.+2
0074  0012  CONST,12;
0075  3720  3720;
0076  0000  0000

0077  0000  DETM,0;
0100  0000  0;
0101  0000  0
0102  0000  DETA1,0;
0103  0000  0;
0104  0000  0
0105  0000  DETA2,0;
0106  0000  0;
0107  0000  0
0110  0000  DETA3,0;
0111  0000  0;
0112  0000  0
0113  0000  C,0;
0114  0000  0;
0115  0000  0
0116  0000  K,0;
0117  0000  0;
0120  0000  0
0121  0000  SUMY,0;
0122  0000  0;
0123  0000  0
0124  0000  BUF2,0;
0125  0000  0;
0126  0000  0
0127  0000  SUMX3,0;
0130  0000  0;
0131  0000  0
```

```
0132  0000  SUMX4,0;
0133  0000  0;
0134  0000  0
0135  0000  SUMX2Y,0;
0136  0000  0;
0137  0000  0
0140  0526  START5,STARTP
0141  0001  ONE,0001
0142  0036  NPP,36
0143  0001  MP,1
0144  0000  TALLY,0
0145  6037  TALLYP,-1741
0146  0000  MPP,0
0147  0414  STAPP2,START2;
            *200
0200  6046  TLS
0201  1204  TAD KUSR
0202  3605  DCA I INTABLE
0203  5206  JMP .+3
0204  4200  KUSR,BCDBIN
0205  7246  INTABLE,7246
0206  1005  TAD TESTP
0207  3003  DCA TEST
0210  1006  TAD APP
0211  3072  DCA AP
0212  1145  TAD TALLYP
0213  3144  DCA TALLY
0214  1142  TAD NPP
0215  3070  DCA NP
0216  1146  TAD MPP
0217  3143  DCA MP
            /ENABLES USER TO SET 1)SAMPLE WIDTH 2)MAX. SIGMA WI
            /3)STARTING CHANNEL, 4)NUMBER OF CHANNELS TO BE SUM
0220  7404  OSR
0221  7440  SZA
0222  5270  JMP START
0223  6032  KCC
0224  4407  JMS I 7
0225  0011  FIN
0226  7000  FNOR
0227  0013  FFIX
0230  0000  FEXT
0231  1044  TAD 44
0232  3070  DCA NP
0233  4407  JMS I 7
0234  0011  FIN
0235  7000  FNOR
0236  0013  FFIX
```

```
0237  0000  FEXT
0240  1044  TAD 44
0241  7041  CIA
0242  3003  DCA TEST
0243  4407  JMS I 7
0244  0011  FIN
0245  7000  FNOR
0246  0013  FFIX
0247  6020  FPUT BUF1
0250  0000  FEXT
0251  1020  TAD BUF1
0252  3143  DCA MP
0253  1020  TAD BUF1
0254  1020  TAD BUF1
0255  1006  TAD APP
0256  3072  DCA AP
0257  4407  JMS I 7
0260  0011  FIN
0261  7000  FNOR
0262  0013  FFIX
0263  0000  FEXT
0264  1044  TAD 44
0265  7041  CIA
0266  3144  DCA TALLY
            /START MAIN INITIALIZATION
0267  7300  CLA CLL
0270  1143  START, TAD MP
0271  3026  DCA M
0272  1072  TAD AP
0273  3023  DCA A
0274  1070  TAD NP
0275  3071  DCA N
0276  1073  TAD STOREP
0277  3010  DCA STORE
0300  1070  TAD NP
0301  7041  CIA
0302  3024  DCA CTR
0303  4407  JMS I 7
0304  7011  FCDF 10
0305  5423  FGET I A
0306  0015  FCON
0307  0014  FLOT
0310  3074  FMPY CONST
0311  6020  FPUT BUF1
0312  0023  FISZ A
0313  5423  FGET I A
0314  7001  FCDF 00
0315  0015  FCON
```

```
0316  0014  FLOT
0317  1020  FADD BUF1
0320  6410  FPUT I  STORE
0321  7010  FNOP
0322  0023  FISZ A
0323  0024  FISZ CTR
0324  0271  FJMP START+1
0325  0000  FEXT
0326  1067  TAD NEG6
0327  3025  DCA CTR2
0330  1073  TAD STOREP
0331  3010  DCA STORE
0332  1066  TAD NEG3
0333  3024  DCA CTR
0334  1026  TAD M
0335  3027  DCA X
0336  4407  JMS I  7
0337  5030  FGET ZERO
0340  0014  FLOT
0341  6063  FPUT SUMX;
0342  6034  FPUT SUMX2;
0343  6031  FPUT SUMXY
0344  6121  FPUT SUMY
0345  5410  START1, FGET I STORE
0346  7010  FNOP
0347  7050  FSZA
0350  0007  FLOG
0351  6124  FPUT BUF2
0352  5027  FGET X
0353  0014  FLOT
0354  6020  FPUT BUF1
0355  3124  FMPY BUF2
0356  1031  FADD SUMXY;
0357  6031  FPUT SUMXY
0360  5020  FGET BUF1
0361  1063  FADD SUMX;
0362  6063  FPUT SUMX
0363  5020  FGET BUF1;
0364  0001  FSQU
0365  1034  FADD SUMX2;
0366  6034  FPUT SUMX2
0367  5124  FGET BUF2
0370  1121  FADD SUMY;
0371  6121  FPUT SUMY
0372  0027  FISZ X
0373  0025  FISZ CTR2
0374  0024  FISZ CTR
0375  0345  FJMP START1
```

```
0376  0000  FEXT
0377  1027  TAD X
0400  1071  TAD N
0401  1067  TAD NEG6
0402  3027  DCA X
0403  1010  TAD STORE
0404  1071  TAD N;
0405  1071  TAD N;
0406  1071  TAD N
0407  1067  TAD NEG6;
0410  1067  TAD NEG6;
0411  1067  TAD NEG6
0412  3010  DCA STORE
0413  4407  JMS I 7
0414  5410  START2, FGET I STORE
0415  7010  FNOP
0416  7050  FSZA
0417  0007  FLOG
0420  6124  FPUT BUF2
0421  5027  FGET X
0422  0014  FLOT
0423  6020  FPUT BUF1
0424  3124  FMPY BUF2
0425  1031  FADD SUMXY;
0426  6031  FPUT SUMXY
0427  5020  FGET BUF1
0430  1063  FADD SUMX;
0431  6063  FPUT SUMX
0432  5020  FGET BUF1;
0433  0001  FSQU
0434  1034  FADD SUMX2;
0435  6034  FPUT SUMX2
0436  5124  FGET BUF2
0437  1121  FADD SUMY;
0440  6121  FPUT SUMY
0441  0027  FISZ X
0442  0025  FISZ CTR2
0443  0547  FJMP I STARP2
0444  5063  START3, FGET SUMX
0445  0001  FSQU
0446  6020  FPUT BUF1
0447  5067  FGET NEG6
0450  0014  FLOT
0451  0010  FNEG
0452  3034  FMPY SUMX2
0453  2020  FSUB BUF1
0454  6077  FPUT DETM
0455  5063  FGET SUMX
```

```
0456  3031  FMPY SUMXY
0457  6020  FPUT BUF1
0460  5121  FGET SUMY
0461  3034  FMPY SUMX2
0462  2020  FSUB BUF1
0463  6102  FPUT DETA1
0464  5063  FGET SUMX
0465  3121  FMPY SUMY
0466  6020  FPUT BUF1
0467  5067  FGET NEG6
0470  0014  FLOT;
0471  0010  FNEG
0472  3031  FMPY SUMXY
0473  2020  FSUB BUF1
0474  4077  FDIV DETM
0475  7010  FNOP
0476  6116  FPUT K
0477  5102  FGET DETA1
0500  4077  FDIV DETM
0501  0006  FEXP
0502  7010  FNOP
0503  6113  FPUT C
            PAUSE

            /CALS MN. COEFS
0504  0000  FEXT
0505  1071  TAD N
0506  7041  CIA
0507  3024  DCA CTR
0510  1073  TAD STOREP;
0511  3010  DCA STORE
0512  1026  TAD M;
0513  3027  DCA X
0514  4407  JMS I 7
0515  5030  FGET ZERO
0516  0014  FLOT
0517  6063  FPUT SUMX
0520  6034  FPUT SUMX2
0521  6127  FPUT SUMX3
0522  6132  FPUT SUMX4
0523  6121  FPUT SUMY
0524  6031  FPUT SUMXY
0525  6135  FPUT SUMX2Y
0526  5027  STARTP, FGET X
0527  0014  FLOT
0530  7010  FNOP
0531  6020  FPUT BUF1
0532  5410  FGET I STORE
```

```
0533  7010  FNOP
0534  6124  FPUT BUF2
0535  5020  FGET BUF1
0536  3116  FMPY K
0537  0006  FEXP
0540  3113  FMPY C
0541  7010  FNOP
0542  0010  FNEG
0543  1124  FADD BUF2
0544  7010  FNOP
0545  0001  FSQU
0546  7010  FNOP
0547  7050  FSZA
0550  0007  FLOG
0551  7010  FNOP
0552  6124  FPUT BUF2
0553  7010  FNOP
0554  5020  FGET BUF1
0555  7010  FNOP
0556  1063  FADD SUMX ;
0557  6063  FPUT SUMX
0560  5020  FGET BUF1;
0561  0001  FSQU
0562  7010  FNOP
0563  1034  FADD SUMX2;
0564  6034  FPUT SUMX2
0565  5020  FGET BUF1;
0566  0001  FSQU;
0567  3020  FMPY BUF1
0570  7010  FNOP
0571  1127  FADD SUMX3;
0572  6127  FPUT SUMX3
0573  5020  FGET BUF1 ;
0574  0001  FSQU;
0575  0001  FSQU
0576  7010  FNOP
0577  1132  FADD SUMX4;
0600  6132  FPUT SUMX4
0601  5124  FGET BUF2;
0602  7010  FNOP
0603  1121  FADD SUMY;
0604  6121  FPUT SUMY
0605  5124  FGET BUF2;
0606  3020  FMPY BUF1;
0607  7010  FNOP
0610  1031  FADD SUMXY;
0611  6031  FPUT SUMXY
0612  5020  FGET BUF1;
```

```
0613  0001  FSQU;
0614  3124  FMPY BUF2;
0615  7010  FNOP
0616  1135  FADD SUMX2Y;
0617  6135  FPUT SUMX2Y
0620  7010  FNOP
0621  0027  FISZ X;
0622  0024   FISZ CTR
0623  0540  FJMP I STARTS
            /DETM
0624  5071  FGET N
0625  0014  FLOT
0626  6113  FPUT C
0627  5034  FGET SUMX2;
0630  7010  FNOP;
0631  3132  FMPY SUMX4;
0632  6020  FPUT BUF1
0633  5127  FGET SUMX3;
0634  7010  FNOP;
0635  0001  FSQU;
0636  0010  FNEG;
0637  1020  FADD BUF1
0640  3113  FMPY C;
0641  6077  FPUT DETM
0642  5063  FGET SUMX;
0643  7010  FNOP;
0644  3132  FMPY SUMX4;
0645  6020  FPUT BUF1
0646  5127  FGET SUMX3;
0647  3034  FMPY SUMX2;
0650  0010  FNEG;
0651  1020  FADD BUF1
0652  3063  FMPY SUMX;
0653  0010  FNEG;
0654  1077  FADD DETM;
0655  6077  FPUT DETM
0656  5063  FGET SUMX;
0657  3127  FMPY SUMX3;
0660  6020  FPUT BUF1
0661  5034  FGET SUMX2;
0662  7010  FNOP;
0663  0001  FSQU;
0664  0010  FNEG;
0665  1020  FADD BUF1
0666  3034  FMPY SUMX2;
0667  1077  FADD DETM;
0670  6077  FPUT DETM
0671  7010  FNOP
0672  0401  FJMP I REPP
```

```
                *2000
                /CALCULATES DETA1
2000    5034    REP; FGET SUMX2;
2001    3132    FMPY SUMX4;
2002    6020    FPUT BUF1
2003    5127    FGET SUMX3;
2004    0001    FSQU;
2005    0010    FNEG;
2006    1020    FADD BUF1
2007    3121    FMPY SUMY;
2010    6102    FPUT DETA1
2011    5031    FGET SUMXY;
2012    7010    FNOP
2013    3132    FMPY SUMX4;
2014    6020    FPUT BUF1
2015    5127    FGET SUMX3;
2016    3135    FMPY SUMX2Y;
2017    0010    FNEG;
2020    1020    FADD BUF1
2021    3063    FMPY SUMX;
2022    0010    FNEG;
2023    1102    FADD DETA1;
2024    6102    FPUT DETA1
2025    5031    FGET SUMXY;
2026    3127    FMPY SUMX3;
2027    6020    FPUT BUF1
2030    5034    FGET SUMX2;
2031    3135    FMPY SUMX2Y;
2032    0010    FNEG;
2033    1020    FADD BUF1
2034    3034    FMPY SUMX2;
2035    1102    FADD DETA1;
2036    6102    FPUT DETA1
2037    7010    FNOP
                /CALCULATES DETA2
2040    5031    FGET SUMXY;
2041    7010    FNOP;
2042    3132    FMPY SUMX4;
2043    6020    FPUT BUF1
2044    5127    FGET SUMX3;
2045    7010    FNOP;
2046    3135    FMPY SUMX2Y;
2047    0010    FNEG;
2050    1020    FADD BUF1
2051    3113    FMPY C;
2052    6105    FPUT DETA2
2053    5063    FGET SUMX;
2054    7010    FNOP;
```

```
2055   3132   FMPY SUMX4;
2056   6020   FPUT BUF1
2057   5034   FGET SUMX2;
2060   7010   FNOP;
2061   3127   FMPY SUMX3;
2062   0010   FNEG;
2063   1020   FADD BUF1
2064   3121   FMPY SUMY;
2065   0010   FNEG;
2066   1105   FADD DETA2;
2067   6105   FPUT DETA2
2070   5063   FGET SUMX;
2071   3135   FMPY SUMX2Y;
2072   6020   FPUT BUF1
2073   5034   FGET SUMX2;
2074   3031   FMPY SUMXY;
2075   0010   FNEG;
2076   1020   FADD BUF1
2077   3034   FMPY SUMX2;
2100   1105   FADD DETA2;
2101   6105   FPUT DETA2;
2102   7010   FNOP
              /CALCULATES DETA3
2103   5034   FGET SUMX2;
2104   3135   FMPY SUMX2Y;
2105   6020   FPUT BUF1
2106   5031   FGET SUMXY;
2107   3127   FMPY SUMX3;
2110   0010   FNEG;
2111   1020   FADD BUF1
2112   3113   FMPY C;
2113   6110   FPUT DETA3
2114   5063   FGET SUMX;
2115   3135   FMPY SUMX2Y;
2116   6020   FPUT BUF1
2117   5031   FGET SUMXY;
2120   3034   FMPY SUMX2;
2121   0010   FNEG;
2122   1020   FADD BUF1
2123   3063   FMPY SUMX;
2124   0010   FNEG;
2125   1110   FADD DETA3;
2126   6110   FPUT DETA3
2127   5063   FGET SUMX;
2130   3127   FMPY SUMX3;
2131   6020   FPUT BUF1
2132   5034   FGET SUMX2;
2133   0001   FSQU;
```

```
2134  0010  FNEG;
2135  1020  FADD BUF1
2136  3121  FMPY SUMY;
2137  1110  FADD DETA3;
2140  6110  FPUT DETA3;
2141  7010  FNOP
2142  5071  FGET N
2143  5110  FGET DETA3;
2144  4077  FDIV DETM;
2145  0010  FNEG;
2146  6020  FPUT BUF1
2147  5141  FGET ONE;
2150  0014  FLOT;
2151  4020  FDIV BUF1;
2152  0002  FSQR;
2153  6110  FPUT DETA3;
2154  7010  FNOP
            /SIGMA IS STORED IN DETA1
2155  5105  FGET DETA2;
2156  4077  FDIV DETM;
2157  6020  FPUT BUF1
2160  0000  FEXT;
2161  1141  TAD ONE;
2162  1141  TAD ONE;
2163  3023  DCA A;
2164  4407  JMS I 7
2165  5023  FGET A;
2166  0014  FLOT;
2167  6124  FPUT BUF2
2170  5110  FGET DETA3;
2171  0001  FSQU;
2172  3020  FMPY BUF1;
2173  4124  FDIV BUF2;
2174  6105  FPUT DETA2
2175  7010  FNOP
            /MU IS STORED IN DETA2.
2176  0000  FEXT
2177  1141  TAD ONE;
2200  1141  TAD ONE
2201  3124  DCA BUF2
2202  4407  JMS I 7
2203  5124  FGET BUF2
2204  0014  FLOT
2205  6124  FPUT BUF2
2206  5030  FGET ZERO
2207  5102  FGET DETA1;
2210  4077  FDIV DETM;
2211  6063  FPUT SUMX
```

```
2212  5105  FGET DETA2;
2213  4110  FDIV DETA3;
2214  0001  FSQU;
2215  1063  FADD SUMX
2216  4124  FDIV BUF2;
2217  0006  FEXP;
2220  6102  FPUT DETA1;
2221  7010  FNOP
2222  5110  FGET DETA3
2223  7000  FNOR;
2224  0013  FFIX;
2225  6020  FPUT BUF1
2226  0000  FEXT
2227  1020  TAD BUF1;
2230  7041  CIA
2231  1003  TAD TEST;
2232  7510  SPA
2233  5235  JMP ZA
2234  5237  JMP ZO
2235  4407  ZA,JMS I 7
2236  2252  FJMP RESET
2237  4407  ZO,JMS I 7
2240  5110  FGET DETA3
2241  0012  FOUT
2242  5105  FGET DETA2
2243  0012  FOUT
2244  5102  FGET DETA1
2245  0012  FOUT
2246  0000  FEXT
2247  4402  JMS I CRLFP
2250  4402  JMS I CRLFP
2251  4407  JMS I 7
2252  0072  RESET,FISZ AP
2253  0072  FISZ AP
2254  7010  FNOP;
2255  7010  FNOP;
2256  7010  FNOP;
2257  7010  FNOP;
2260  7010  FNOP;
2261  7010  FNOP
2262  0143  FISZ MP;
2263  7010  FNOP;
2264  7010  FNOP;
2265  7010  FNOP
2266  0000  FEXT;
2267  2144  ISZ TALLY;
2270  5272  JMP .+2;
2271  7402  HLT
2272  5673  JMP I BEGINP
2273  0270  BEGINP,START
```

```
                /CALCULATES BINARY NUMBERS FROM BCD NUMBERS
                *4200
4200    0000    BCDBIN,0
4201    1634    TAD I AC
4202    3233    DCA TEMPH
4203    1233    TAD TEMPH
4204    0230    AND LDIGIT
4205    7112    CLL RTR
4206    3232    DCA COUNT
4207    1232    TAD COUNT
4210    7010    RAR
4211    1232    TAD COUNT
4212    7041    CMA IAC
4213    1233    TAD TEMPH
4214    3233    DCA TEMPH
4215    1233    TAD TEMPH
4216    0231    AND MDIGIT
4217    7112    CLL RTR
4220    3232    DCA COUNT
4221    1232    TAD COUNT
4222    7010    RAR
4223    1232    TAD COUNT
4224    7041    CMA IAC
4225    1233    TAD TEMPH
4226    3634    DCA I AC
4227    5600    JMP I BCDBIN
4230    7400    LDIGIT,7400
4231    7760    MDIGIT,7760
4232    0000    COUNT,0
4233    0000    TEMPH,0
4234    0044    AC,44
4235    0000    CRLF,0
4236    7300    CLA CLL
4237    1244    TAD K215
4240    4246    JMS TYPE
4241    1245    TAD K212
4242    4246    JMS TYPE
4243    5635    JMP I CRLF
4244    0215    K215,215
4245    0212    K212,212
4246    0000    TYPE,0
4247    6041    TSF
4250    5247    JMP .-1
4251    6046    TLS
4252    7300    CLA CLL
4253    5646    JMP I TYPE
```

## APPENDIX II

## PROGRAM TO ANALYZE PULSER SPECTRA

```
                  FIXMPI   FJMP=0000
                  FIXMRI   FJMS=7000
                  FISZ=0000
                  FEXT=0000
                  FSQU=0001
                  FSQR=0002
                  FSIN=0003
                  FCOS=0004
                  FATN=0005
                  FEXP=0006
                  FLOG=0007
                  FNEG=0010
                  FIN=0011
                  FOUT=0012
                  FFIX=0013
                  FLOT=0014
                  FNOR=7000
                  FCDF=7001
                  FSW0=7002
                  FSW1=7003
                  FHLT=7004
                  FSMA=7110
                  FSZA=7050
                  FSPA=7100
                  FSNA=7040
                  FNOP=7010
                  FSKP=7020
                  FCON=0015

                  /     14:11:73 M HILES
                  *20
0020    0000      A,0;
0021    0000      CTR,0;
0022    0000      TALLY,0
                  *200
0200    6046      TLS
0201    1327      TAD KUSB
0202    3730      DCA I INTABLE
0203    4407      JMS I 7
0204    7011      FCDF 10
0205    5324      FGET P
0206    6020      FPUT A
0207    5312      FGET ZERO
0210    6321      FPUT N
0211    6316      FPUT D
```

```
                /(1)
0212   5420   START, FGET I A
0213   0015   FCON
0214   0014   FLOT
0215   3307   FMPY CONST
0216   6304   FPUT TEMSTR
0217   0020   FISZ A
0220   5420   FGET I A
0221   0015   FCON
0222   0014   FLOT
0223   1304   FADD TEMSTR
0224   6304   FPUT TEMSTR
0225   0020   FISZ A
                /(2)
0226   0021   FISZ CTR
0227   0022   FISZ TALLY
0230   7020   FSKP
0231   7004   FHLT
                /(3)
0232   5315   FGET NEG2
0233   0014   FLOT
0234   1304   FADD TEMSTR
0235   7100   FSPA
0236   0212   FJMP START
                /(4)
0237   5304   A1, FGET TEMSTR
0240   1316   FADD D
0241   6316   FPUT D
                /(5,6)
0242   5021   FGET CTR
0243   0014   FLOT
0244   3304   FMPY TEMSTR
0245   1321   FADD N
0246   6321   FPUT N
                /(7)
0247   0021   FISZ CTR
0250   0022   FISZ TALLY
0251   7020   FSKP
0252   7004   FHLT
                /(8)
0253   5420   FGET I A
0254   0015   FCON
0255   0014   FLOT
0256   3307   FMPY CONST
0257   6304   FPUT TEMSTR
0260   0020   FISZ A
0261   5420   FGET I A
0262   0015   FCON
```

```
0263  0014  FLOT
0264  1304  FADD TEMSTR
0265  6304  FPUT TEMSTR
0266  0020  FISZ A
            /(9)
0267  5315  FGET NEG2
0270  0014  FLOT
0271  1304  FADD TEMSTR
0272  7100  FSPA
0273  7020  FSKP
0274  0237  FJMP A1
            /(10)
0275  5321  FGET N
0276  4316  FDIV D
0277  0012  FOUT
            /(12)
0300  5312  FGET ZERO
0301  6316  FPUT D
0302  6321  FPUT N
0303  0212  FJMP START
0304  0000  TEMSTR,0;
0305  0000  0;
0306  0000  0
0307  0012  CONST,12;
0310  3720  3720;
0311  0000  0000
0312  0000  ZERO,0;
0313  0000  0;
0314  0000  0
0315  7776  NEG2,-2
0316  0000  D,0;
0317  0000  0;
0320  0000  0
0321  0000  N,0;
0322  0000  0;
0323  0000  0
0324  2002  P,2002;
0325  0000  0;
0326  6003  -1775
0327  0331  KUSR,BCDBIN
0330  7246  INTABLE,7246
0331  0000  BCDBIN,0
0332  1765  TAD I AC
0333  3364  DCA TEMPH
0334  1364  TAD TEMPH
0335  0361  AND LDIGIT
0336  7112  CLL RTR
0337  3363  DCA COUNT
```

```
0340  1363  TAD COUNT
0341  7010  RAR
0342  1363  TAD COUNT
0343  7041  CMA IAC
0344  1364  TAD TEMPH
0345  3364  DCA TEMPH
0346  1364  TAD TEMPH
0347  0362  AND MDIGIT
0350  7112  CLL RTR
0351  3363  DCA COUNT
0352  1363  TAD COUNT
0353  7010  RAR
0354  1363  TAD COUNT
0355  7041  CMA IAC
0356  1364  TAD TEMPH
0357  3765  DCA I AC
0360  5731  JMP I BCDBIN
0361  7400  LDIGIT,7400
0362  7760  MDIGIT,7760
0363  0000  COUNT,0
0364  0000  TEMPH,0
0365  0044  AC,44
```

## BIBLIOGRAPHY

(1) Butkov E., Mathematical Physics (Addison Wesley Publishing Company, Don Mills, 1964)

(2) Kelly l.g., Handbook of Numerical Methods and Applications (Addison Wesley Publishing Company, Don Mills, 1967)

(3) Sokolnikoff I.S. and Redheffer R.M., Mathematics of Physics and Modern Engineering (McGraw-Hill Book Company, Toronto, 1966) Chap. 10, p. 673.

(4) Kipling A., MCA Service Program (Unpublished)

(5) Forsythe G., J. Soc. Indust. Appl. Math., 5 (Jan. 1958)

(6) Digital Equipment Corporation, Introduction To Programming 1972 (Digital Equipment Corporation, Maynard, Mass., 1972)